MIDI TO CV CONVERTER

MIDI-2CV_   MIICV

GROOVE (viii) 1988

Groove Electronics,
Unit 30,
Barnack Industrial Centre,
Kingsway Trading Estate,
Wilton,
WILTSHIRE,
SP2 0AW.

tel:(0722) 743712

1. INTRODUCTION

Congratulations you are now the proud owner of a marvel of modern technology.......the GROOVE MIDI-2CV converter. It will enable you to use those wonderful old monosynths, with their warm and often unpredictable tonal qualities to bring new life into your music.

The Groove M2CV provides a control voltage suitable for driving monosynths which use the 1V/Oct scaling system, this system is used by the majority of synths.
In its most basic form the MIDI-2CV_ will control a single mono-synth, providing Control Voltage, Trigger, and VCA/VCF Level as its outputs. In addition pitch-bend (+/- Oct.) is added to the CV output, the amount being variable from 0 to +/-1 Oct. The VCA/VCF Level output may be assigned to key-velocity, modulation-wheel, after-touch, breath-controller......The Trigger output may be set to cover most eventualities, including S-trigger. A 'Coarse Tune' trimmer is provided to adjust the Control Voltage by +/-2 Octaves but since the trimmer is of the multi-turn variety it can also be used for fine tuning!

To this basic unit the following options may be added;

1:**SECOND CV**, another completely independant channel to control a second mono-synth, this second channel provides all the features listed above and may be set on a different MIDI channel.

2:**MEMORY**, this gives the M2CV the ability to store various set-ups in memory for instant recall, these will be retained after power-down. The basic M2CV does not have such memory and will always default to the factory-set mode on power-up.

3:**WASP**, an interface for the WASP synthesiser, it may be set on a different MIDI channel and provides note on/off commands to the WASP.

4:**DIN-SYNC**, allows sychronisation of old Roland drum machines and sequencers to the MIDI clock.

5:**ARP-CLK**, this provides a trigger output derived from the MIDI clock signal, typically this would be used for triggering the arpeggiator or sequencer available on some synths.

6:**DRUM-TRIG**, a software update that enables the M2CV to play Simmons type drum synth modules, 6 channels are available if the 2ndCV is fitted to the M2CV.

2. PRINCIPLES OF OPERATION

Before describing how to use the Groove M2CV we present here an introduction to the control voltage system used on older synths, for the benefit of those of you who are not familiar with it.

In the days before MIDI sockets, to provide for external control of synths it was necessary to provide separate control inputs for the various elements of the sound that must be controlled. So some means of controlling the pitch of the sound was required as was control over the duration of the sound ie. a gate signal. These two signals were the two basic control inputs provided on most synths in addition some means of controlling the filter cut-off was often provided. Unfortunately in the days before MIDI there was no standardisation of synth control and so several types of trigger (gate) signal and two forms of pitch control signal were adopted by the various manufacturers.

The two types of pitch control signal employ different scaling, and it is only possible to cater for one type of scaling. The '1Volt-per-Octave' (1V/Oct) scaling, so called because for each Volt increase the pitch increases by one octave, is used in the Groove M2CV as this is the most commonly found scaling. The other type of scaling is known as 'Hertz-per-Volt' (Hz/V), actually it is kiloHertz per Volt, and for each Volt increase the pitch increases by 1 kiloHertz. The Hz/V system was used extensively by Yamaha on their monosynths, so sadly these will not track the pitch control signal produced by the M2CV. Fortunately most of the better known monosynths employ the V/Oct scaling and will work with the M2CV.

There are basically four types of trigger signal all of which the M2CV can be set to produce. The odd one out of the four is the so called S-trigger, the 'S' stands for 'short', this system uses a relay to make a contact when a trigger is required. The S-trigger is not a standard feature of the M2CV but is available as an add-on, this trigger system is the rarest of the four. The other three trigger systems use a voltage level to determine trigger on/off these trigger systems are:

1. POSITIVE GOING (+VE) TRIGGER :output normally sits at 0V and goes to a positive voltage for the length of the gate period.
2. GROUND TRIGGER (GND) :output normally sits at a positive voltage and goes to 0V for the gate.
3. NEGATIVE GOING (-VE) TRIGGER :output normally sits at 0V and goes to a negative voltage for the gate.

As mentioned all of these three trigger systems are available on the M2CV as standard so there should be no problems in triggering any monosynth.

The third control output provided by the Groove M2CV is intended for controlling the filter cut-off frequency of the synth. Control of this synth parameter, by external means, is not so common and so there is no predetermined format for this signal. So the M2CV provides a control signal in the range 0 to +8V and this may be controlled by MIDI velocity, mod-wheel, after-touch or breath controller data. If your synth does not have a control input for VCF cut-off then we can probably provide it with one to ensure you get the full benefit of this useful control output.

3. CONNECTIONS

This section describes the connections to the M2CV and how to link up your synth to the M2CV.

The diagram below illustrates the rear panel of the M2CV, it has two 5-pin DIN sockets for MIDI IN and THRU, eight 3.5mm jack sockets provide the control output signals.

power S-Trig level arp-clk level MIDI MIDI
on off trig pitch trig pitch THRU IN

SYNTH B SYNTH A

The three synth control outputs are PITCH, LEVEL and TRIG, there are two sets, for synth A and synth B. The synth B outputs will only function if your M2CV has the 2nd.CV option fitted. These three outputs should be linked as follows;

PITCH: This is the pitch control output and should be linked to the pitch control input on your synth, this should be marked 'CV IN' however there are other possibilities, if in doubt consult your synth manual.

LEVEL: This is the VCF cut-off control output and should be linked to the FILTER control input on your synth.

TRIG : This is the trigger output and should be linked to the trigger input on your synth, generally labelled TRIGGER or GATE IN.

The other two outputs are available as options, these are;

S-TRIG: This is the S-trigger control output as previously described, it should be connected to the synth trigger input in place of the TRIG output above. The S-trigger is only available for one of the synths the M2CV is controlling, this is synth A by default but may be wired to synth B if required.

ARP-CLK: This is the Arpeggiator Clock output and provides a 0<->10V clock signal derived from the MIDI clock and would typically be used for syncing an arpeggiator or sequencer to the MIDI clock. So this output should be linked to an input marked 'EXT CLOCK' or similar. For more details about this output see section 5.2.2

Two more optional outputs may appear on the rear of the M2CV unit and are labelled **WASP** and **DIN-SYNC**. The WASP connection is a 7-pin DIN socket and should be linked to the 7-pin DIN on the Wasp it is controlling. The DIN-SYNC connection is a 5-pin DIN socket that looks like a MIDI socket and should be linked to DIN-SYNC IN of the unit you wish to sync to MIDI.

4. SETTING UP

Having linked your synth up to the M2CV it will be necessary to callibrate the scaling of PITCH output from the M2CV. This should only need to be done once for any given synth, depending on how stable the voltage-to-pitch converter within the synth is. Allow the synth and M2CV to warm up for 10 minutes before attempting to callibrate the V/Oct scaling.

As you look at the M2CV from the front there are 6 small holes down the left hand end-cheek, as shown below;

V/Oct Bend Tune
Tune V/Oct Bend

Synth A Synth B

You will need a small screwdriver (or tweeker) to adjust these trimpots, there are three trimpots per synth control channel and these function as follows;

V/Oct : This is the most critical of the three as this sets the pitch scaling of the M2CV and if this is not right at the start the synth will only stay in tune over a small range of the keyboard. This is a multiturn pot to allow the V/Oct to be set accurately, it provides about +/-20% variation so the V/Oct may be set between 0.8 and 1.2 V/Oct. This range is necessary since the scaling within the synth itself will drift over the years.

Tune : This is another multiturn pot which provides a total tune range of +/-2 Octaves but is sufficiently fine to allow for fine tuning as well.

Bend : This is a normal, single turn pot and is used to set the amount of pitch-bend which may be applied to the pitch output. Pitch-bend may be set form 0 to +/-1 Oct.

4.1 Callibrating V/Oct :

In order to perform this callibration it will be necessary to connect a MIDI keyboard to the MIDI IN on the M2CV, you may also have to proceed to the next section, Editing Parameters, first to set which MIDI channel synth A or B responds on. Assuming that your synth is responding to the MIDI keyboard proceed as follows;

1. Set the synth and MIDI keyboard to produce a sine wave or similar pure tone that is suitable for tuning. It will be necessary to hear both synth and MIDI keyboard as you will be tuning the synth to the MIDI keyboard.
2. Centre any tune controls on the synth and MIDI keyboard.
3. Play Middle C (MIDI note# 60) on the MIDI keyboard.
4. Adjust 'Tune' trimpot on the M2CV until both keyboards are in tune.
5. Play a note 3-4 octaves up (C6-C7) on the MIDI keyboard.
6. Adjust 'V/Oct' trimpot on the M2CV until both keyboards are in tune.

5. EDITING PARAMETERS

The Groove MIDI-2CV has been designed with maximum flexibility in mind, as a consequence of this there are many parameters which may be edited. This section provides a detailed description of all the parameters which may be edited.

The display consists of eight LEDs which are used to indicate the parameter you are editing and its value. These LEDs may be in one of four conditions; off, on, slow-flashing, fast-flashing, for the purpose of this description these conditions will be indicated by the following symbols;

1.OFF	=0
2.ON	=1
3.SLOW-FLASH	=+
4.FAST-FLASH	=*
UNKNOWN	=x

The last entry above, 'UNKNOWN', refers to LEDs whose status depends on the value they are displaying, hence these LEDs will be either ON or OFF.

Thus the condition of the LED display will be indicated by the following notation: LED{+OO*xxxx}, the eight LEDs are depicted left to right as they are on the front panel, corresponding to the eight switches A-H.

In describing the switch pushes required the following notation will be used [A], [B],.....,[H], where [A] refers to switch A !

There are four basic modes of operation;

1. MEMORY RECALL
2. DEVICE SELECT
3. EDIT MODE
4. MEMORY STORE

To step from one mode to the next use key [A], after mode 4 the system returns to mode 1. At power-on the M2CV enters mode 1 and recalls the memory which was active prior to power-off if battery backed RAM is present, otherwise the M2CV is set-up according to a set of parameters held in ROM.
EXAMPLE: power has just been applied, now step through the modes.

power-on	-> LED {01001101}	:RECALL MODE, mem. #13
KEY [A]	-> LED {10010001}	:DEVICE SELECT = SYNTH A
KEY [A]	-> LED {+0000000} :	:EDIT MODE
KEY [A]	-> LED {*0101101}	:MEMORY STORE, mem. #13
KEY [A]	-> LED {01001101}	:RECALL MODE, mem. #13

So the pattern continues, stepping through the four modes.

In general the editing process uses the keys/leds as follows;

KEY/LED A   steps between modes of operation.
KEY/LED B,C,D   selects parameter to edit.
KEY/LED E,F,G,H alters parameter value.

Parameter values will be in one of two forms;

(i) Binary Values

In many cases the parameter value will be a number in the range of 0 to 15 and in order to display this on LEDs the binary number system is used;

LED E	F	G	H	VALUE
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	0	0	0	8
1	0	0	1	9
1	0	1	0	10
1	0	1	1	11
1	1	0	0	12
1	1	0	1	13
1	1	1	0	14
1	1	1	1	15

To alter the value of the parameter simply press one of the keys [E]-[H] and the value of that particular 'bit' will flip between one and zero. This form of representation is used for setting the MIDI channel, for instance, where a value from 1 to 16 is needed.

(ii) Flag Values

For many parameters it is only necessary to select one (or more) of several options, in this case each of the keys [E]-[H] is used to choose one of the options. In some cases the options are mutually exclusive, for instance it is only possible to select one type of trigger output from the three available. On the other hand it is possible to select any combination of four sources of MIDI data to control the LEVEL output.

5.1 MEMORY RECALL :   LED {0100xxxx} (xxxx=current mem. #)

this is only applicable to systems with battery backed RAM, this provides 16 memory locations in which to store M2CV set-ups. At power-on the M2CV will recall the memory that was active before the M2CV was last turned off. This memory number is displayed on LEDs E-H in binary form, LED B will also be lit to indicate memory recall mode (also indicated by LED A = OFF!). So a typical display after power on is {01001101} showing that memory number 13 has been recalled.

To recall another memory proceed thus:

1. select memory number with keys [E] to [H], setting the desired memory number 0-15 in binary. Note that when the mem. number displayed is not the same as that currently active then LED B will flash slowly.
2. having selected the desired mem. number press key [B], the new parameters are installed.

If LED C flashes rapidly then the mem. just recalled has invalid data in it, this may be the case when the unit has been just been purchased and the memories have not yet been filled with data.

EXAMPLE: mem. #3 currently active and you want mem. #11.

```
LED {01000011}                      :mem. # 3 currently active
KEY [E] -> LED {0+001011}           :mem. #11 set
KEY [B] -> LED {01001011}           :mem. #11 recalled OK.
        -> LED {01*01011}           :mem. #11 data corrupted.
```

5.2 DEVICE SELECT :   LED {10010xxx} (xxx=current device)

This mode is named Device Select since it is used to select which of the three devices (Synth A, Synth B or Wasp) will have its parameters edited when in Edit Mode.

The device is selected thus;

```
KEY [H] -> LED {10010001}           :Selects Synth A
KEY [G] -> LED {10010010}           :Selects Synth B
KEY [F] -> LED {10010100}           :Selects Wasp
```

However selecting a device for editing is not the only function performed in this mode, two more functions are performed. So to select between the three sub-modes within the Device Select mode use keys [B], [C] and [D] thus;

```
KEY [B] -> LED {1100xxxx}           :Arp. Edit
KEY [C] -> LED {1010x0xx}           :Key Transpose
KEY [D] -> LED {10010xxx}           :Device Select
```

5.2.1 KEY TRANSPOSE : LED {1010x0xx} (x0xx=transpose value)

This function is used to move the 6 octave range of the M2CV around the MIDI range of 10.5 octaves. It is possible to 'transpose' the M2CV range by +/- 3 octaves in 1 octave steps and each of the three devices (Synth A, Synth B and Wasp) have their own transpose value. Thus the device is selected as in 5.2 above then the transpose value altered with keys [E], [G] and [H]. Key [E] selects whether the transpose is up or down and keys [G], [H] select the number of octaves (in binary) from 0 to 3.

EXAMPLE: here are some examples of transpose values;

DISPLAY	TRANSPOSE	KEY RANGE
LED {10100000} OR LED {10101000}	:no transpose	24 - 96
LED {10101010}	:down 2 oct.	48 -120
LED {10100001}	:up 1 oct.	12 - 84
LED {10100011}	:up 3 oct.	0 - 60
LED {10101001}	:down 1 oct.	36 -108

Note that in transposing upwards the M2CV range moves down the MIDI key range and vice-versa.

5.2.2 ARP. EDIT : LED {1100xxxx} (xxxx= arp. clk mode)

This mode is provided for editing the arpeggiator clock output, obviously it is only applicable to systems with the Arp. Clk option fitted. This output would typically be used for synchronising an arpeggiator or sequencer to the MIDI clock, it supports MIDI START, STOP, CONTINUE and CLOCK data. There are two parameters to edit;

(i) **CLOCK RATE**, this is the number of MIDI clocks per Arp. Clock, ie. MIDI clock divide factor. There are eight values and these are selected with keys [F] to [H] in binary format as follows;

LED {F G H}	CLOCK DIVIDE
0 0 0	2
0 0 1	3
0 1 0	4
0 1 1	6
1 0 0	8
1 0 1	12
1 1 0	16
1 1 1	24

(ii) **CLOCK SENSE**, this determines which direction the Arp. Clock output goes for a clock event, it is altered with key [E] which selects between a HI->LO clock or a LO->HI clock. In both cases the Arp. clock output swings between 0<->+10V. In most cases a LO->HI clock will be used, the state of this parameter is shown on LED E as;

LED {11000xxx}	:clock HI->LO
LED {11001xxx}	:clock LO->HI

EXAMPLES: Here are some examples of Arp. Clock values;

LED {11001000}	:LO->HI, 12 ppqn
LED {11001011}	:LO->HI, 4 ppqn
LED {11000110}	:HI->LO, 1.5 ppqn

5.3 EDIT MODE : LED {+0000000}

This mode is used to edit the parameters of the device selected previously (sect.5.2). The parameter is selected with keys [B] to [D] and each key selects one of two parameters, by pressing a given key more than once editing will step between the two parameters, the LED for the given key flashes slow or fast depending on which parameter is selected.
The parameters are selected as follows;

KEY [B] -> LED {++00xxxx}	:MIDI CHANNEL
KEY [B] -> LED {+*00xxxx}	:KEY PRIORITY
KEY [C] -> LED {+0+0xxxx}	:TRIGGER TYPE
KEY [C] -> LED {+0*0xxxx}	:PITCHBEND SET-UP
KEY [D] -> LED {+00+xxxx}	:LEVEL SOURCE
KEY [D] -> LED {+00*xxxx}	:LEVEL OFFSET

It is possible to move between the parameters in any order with the obvious constraint that the second functions on the keys can only be accessed after the first !!!

5.3.1 MIDI CHANNEL : LED {++00xxxx} (xxxx= MIDI channel)

This is the MIDI channel on which the device will receive data, it is set in binary format with keys [E]-[H], now as binary format only allows for the numbers 0 to 15 it is necessary for you to add 1 to the value when interpreting the display or to subtract 1 when entering a value. So 0000 = chan.1, 0001 = chan.2, 0010 = chan.3,......., 1111 = chan.16, there is NO OMNI mode.
EXAMPLES: here are some typical displays;

LED {++001001}	:MIDI CHANNEL 10
LED {++000000}	:MIDI CHANNEL 1
LED {++001101}	:MIDI CHANNEL 14

5.3.2 KEY PRIORITY : LED {+*00Xxxx} (xxx=priority, X=start)

This selects one of three key priorities for the device, it also selects if there is a clock with start for the Arp Clock or Din-Sync modes. The key priorities are Flag values and are selected with keys [F]-[H], only one priority may be active at a time, selection is as follows;

KEY [H] -> LED {+*00x001}	:NEW PRIORITY
KEY [G] -> LED {+*00x010}	:LOW PRIORITY
KEY [F] -> LED {+*00x100}	:HIGH PRIORITY

NEW PRIORITY: The M2CV will assign the most recent NOTE-ON.
LOW PRIORITY: The M2CV will assign the lowest NOTE-ON.
HIGH PRIORITY:The M2CV will assign the highest NOTE-ON.

Thus LOW priority would be used to make the synth follow the bass line of a polyphonic part and HIGH priority for the lead line. New priority should generally be used for monophonic parts.

CLOCK WITH START : This mysterious parameter has been put in here as there was nowhere else to put it ! It is used to get around the problem that occurs with the MIDI START command, some systems interpret the START as the first CLOCK of the sequence, whereas other systems will send a CLOCK with the START. This parameter enables you to select whether there is a clock with start or not and adjusts the Arp. Clock output accordingly. Key [E] alters this parameter as follows;

KEY [E] -> LED {+*000xxx} :NO CLOCK WITH START
KEY [E] -> LED {+*001xxx} :CLOCK WITH START

This parameter is altered by key [E] regardless of which device is currently selected.

5.3.3 TRIGGER TYPE : LED {+0+0Xxxx} (X=Multi/Single, xxx=trig)

This parameter selects one of three trigger types for the TRIG output and whether notes will be retriggered (multi.) if a new note on is received before the last note was released. The three trigger types were described in section 2, selection is;

KEY [F] -> LED {+0+0X100} :-VE TRIGGER
KEY [G] -> LED {+0+0X010} :GND TRIGGER
KEY [H] -> LED {+0+0X001} :+VE TRIGGER

If you have the S-TRIG option fitted then the +VE trigger should be selected to drive the S-TRIG output correctly.
To select multi/single trigger use key [E], with single trigger the synth will only be triggered if a new note is played and there are no notes currently sounding, select thus;

KEY [E] -> LED {+0+00xxx} :SINGLE TRIGGER
KEY [E] -> LED {+0+01xxx} :MULTI TRIGGER

5.3.4 PITCHBEND SET-UP : LED {+0*0xxxx} (xxxx= bend amount)

This is a utility rather than a parameter, it enables pitchbend amount to be set-up in semitone steps. The bend amount is set in binary format on keys [E] to [H], the amount may vary between 0 semitones -> LED {+0*00000}
and +/- 1 Oct. -> LED {+0*01100}
In order to set-up the pitchbend amount, set the desired amount on keys [E] to [H] in binary format. The device you are working on will be continuously triggering, every 400ms or so, if you notice a difference in pitch between adjacent triggers then BEND amount requires adjustment. Adjust the relevant BEND trimmer (see section 4) until the adjacent triggers are at the same pitch.

5.3.5 LEVEL SOURCE : LED {+00+xxxx} (xxxx= source)

This parameter selects which sources of MIDI data have control over the LEVEL output, there are four to choose from as follows;

KEY [E]	-> LED {+00+1xxx}	:BREATH CONTROLLER
KEY [F]	-> LED {+00+x1xx}	:AFTER TOUCH
KEY [G]	-> LED {+00+xx1x}	:MOD. WHEEL
KEY [H]	-> LED {+00+xxx1}	:VELOCITY

Any combination of the above four may be selected to control the LEVEL output.

5.3.6 LEVEL OFFSET : LED {+00*xxxx} (xxxx= offset)

This parameter is used to set an offset voltage on the LEVEL output, this may be necessary if you are applying the LEVEL output to the filter cutoff on the synth and you want to raise the filter cut-off frequency. A typical situation where this would be required is for after-touch, where when you release all the keys the after-touch value will go to zero, this would normally close the filter right up but with offset you can control the close-up point. The offset value is entered in binary format using keys [E] to [H], the amount of offset ranges from 0 to a value just short of the maximum possible output on LEVEL;

LED {+00*0000}	:NO OFFSET
to	
LED {+00*1111}	:MAX. OFFSET

5.3.7 WASP PARAMETERS :

Although all of the above parameters may be edited on device C, the WASP channel, only a small number have any effect on the WASP. Basically the WASP only responds to NOTE ON/OFF commands so only parameters associated with this have any effect, thus only the following parameters are effective for the WASP;

MIDI CHANNEL	LED {++00xxxx}
KEY PRIORITY	LED {+*00Xxxx}
TRIGGER TYPE	LED {+0+0xxxx}
KEY TRANSPOSE	LED {1010x0xx}

<u>5.4 MEMORY STORE :</u> LED {*010xxxx} (xxxx= mem. #)

Having edited the parameters for a particular synth setup, you may now wish to save this for future use. To use memory store your M2CV must have battery backed RAM installed, assuming that it does proceed as follows:

1. Select memory number from 0 to 15 with keys [E] to [H], if this is different from the current mem.# then LED C will slow-flash.
2. Press key [C], the data is now stored in the mem.# shown on LED E to H and will be retained after power down for future recall.

EXAMPLE: store edited data in mem.#14, not current mem.#.

KEY [E] TO [H]	-> LED {*0+01110}	:SELECT MEM.#14
KEY [C]	-> LED {*0101110}	:DATA STORED

MIDI-2CV MIDI SPECIFICATION

Groove-[illegible] V1.0 Nov. 1988

function	receive	transmit	remarks
Channel	1-16	1-16	Separate channels for three devices
Key number	0-127	no	
Vel. KEY ON KEY OFF	9n 1-127 9n 0 8n 1-127	no no	
After KEY Touch CHAN.	no yes	no no	All controllers and velocity may be routed to LEVEL output
Control 1:Mod Change Wheel 2:Breath	yes yes	no no	
Pitchbend	yes	no	MS 7 bits only
Sys. Real Time Active sense start,stop , clock,cont	 yes yes yes	 no no no	Timing commands used with DIN-SYNC and ARP.CLK option